

# Syncing Success

Managing Online/Offline Game  
States Across Multiple Devices

Dave Geurts and Colter Haycock  
Software Engineers  
Smart Bomb Interactive

The logo for GDC 13 NEXT, featuring the text "GDC" in a large, bold, sans-serif font, with "13" inside a small circle to the right of the "D", and "NEXT" in a smaller, bold, sans-serif font below it. The background is a vibrant blue and green abstract design with various geometric shapes and glowing lines.

**GDC**  
13  
**NEXT**

NOVEMBER 5-7, 2013  
EXPO DATES: NOV 5-6  
LOS ANGELES, CA

[GDCNext.com](http://GDCNext.com)









# TUNNEL TOWN



**SMART BOMB**  
INTERACTIVE



# Goals: Create mobile game that...

- Supports local only mode
- Allows play on multiple devices
- Seamlessly supports online and offline play
- Is non-intrusive when switching devices
- Requires minimal server logic





# Our Recipe



1) Serialized hashed game state



protobuf

Protocol Buffers - Google's data interchange format

2) Security



3) Server with volatile caching and persistent storage



redis



mongoDB

4) Save state system for online and offline play



# 1) Hashed Game State

- Serialization - Google Protocol Buffers

## Pros

- Less Server Code



protobuf

Protocol Buffers - Google's data interchange format

## Traditional Way

### Players

createAccount()  
login()  
visitBuddy()

### Buddy List

addBuddy()  
removeBuddy()  
editBuddy()

### StarGifting

giftStarToBuddy()  
grantStarGift()

### Bunnies

addBunny()  
removeBunny()  
editBunny()

### In App Purchase

verifyPurchase()

### Furniture

addFurniture()  
removeFurniture()  
attachBunny()

### Player

addGems()  
addStars()  
addExperience()  
spendCurrency()

### GemDigs

mineGems()  
restTiredBunnies()

### Inventory

addItem()  
removeItem()  
editItem()  
tradeItem()

### Burrow

markTileDug()  
fillTile()  
decorateTile()

## Serialized Hashed Game State

### Players

createAccount()  
login()  
visitBuddy()

### StarGifting

giftStarToBuddy()  
grantStarGift()

### In App Purchase

verifyPurchase()

### SaveState

saveData()  
loadData()

# Server Code Differences





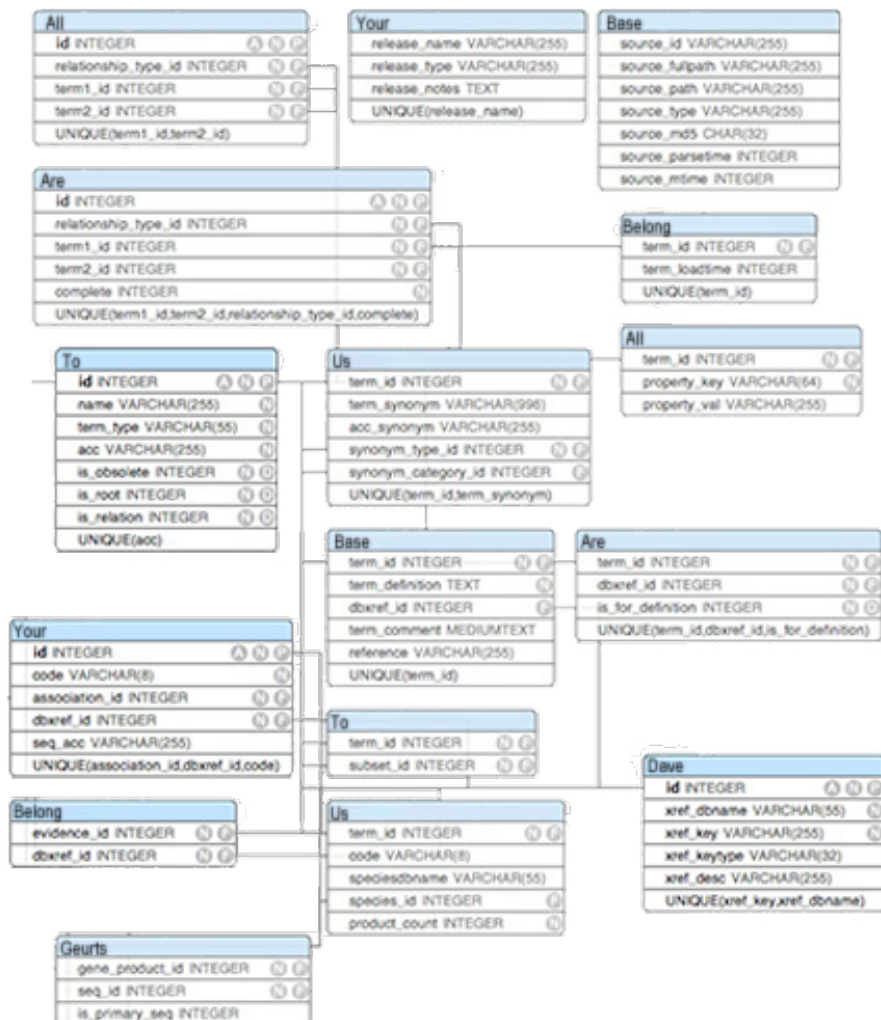
# 1) Hashed Game State

- Serialization - Google Protocol Buffers

## Pros

- Less Server Code
- Fewer Database Schemas to Manage

## Traditional Way Relational



## Serialized Game State Hash NoSQL

### PlayersCollection

```
{
  "_id" : ObjectId("5152224bc33679484192f1f8"),
  "dataObjs" : {
    "GlobalState" : {
      "binary" : "TYecFyx0VfNnzh7S5Gp+4OYOOwJ73iOSlagcdGBFvtq3M9TekT1SY
MKQNHQ8HMQeoSrkl/Ww8yW3/PYRK1V/1Ldtu/eoysSqZvzXJ5CP6FuTYQyroF5L
RyV5YZhozMcYc+jecsy/JZIFRj5RLad9tVaHOEEF/nyiWKK74sWbOLixdx5+hGv0K
nmniVy6LqPfDUddf+Xi4+9LPvSRWpD0ewddAPdufUPg5ZugGf3exYLf4FgMQUY..."
      "playerKey" : "carmenSanDiego",
      "ts" : ISODate("2013-03-26T22:38:30.006Z")
    }
  },
  "lastGemBonus" : ISODate("1970-01-01T00:00:01Z"),
  "lastLogin" : ISODate("2013-03-26T22:33:47.37Z"),
  "lastStarAwarded" : ISODate("1970-01-01T00:00:01Z"),
  "logins" : 1,
  "numStarsToGive" : 0,
  "playerData" : {
    "level" : 1,
    "experience" : 0,
    "gems" : 0,
  }
}
```

## Database Differences



# 1) Hashed Game State

- Serialization - Google Protocol Buffers

## Pros

- Less Server Code
- Fewer Database Schemas to Manage
- Obfuscation



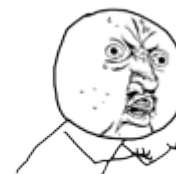
## Traditional Way

```
tunnelTown.php?qId=3&c=saveBurrowData&bunnies=
[{'name':'brownBunny','experience':14,'stars':98,'pos':{'x':23,'y':23,'x1':1}},{'name':'KarlMalone','experience':14,'stars':345,'pos':{'x':23,'y':23,'x1':1}},{'name':'clydeDrexler','experience':2,'stars':12,'pos':{'x':23,'y':23,'x1':1}},{'name':'Michael Jackson','experience':4,'stars':12,'pos':{'x':23,'y':23,'x1':1}},{'name':'Barack Obama','experience':0,'stars':0,'pos':{'x':23,'y':23,'x1':1}},{'name':'2pac','experience':999,'stars':342,'pos':{'x':23,'y':23,'x1':1}},{'type':'wall','name':'clock','pos':{'x':23,'y':23,'x1':1}},{'type':'floor','name':'couch','pos':{'x':23,'y':23,'x1':1}},{'type':'wall','name':'clock','pos':{'x':23,'y':23,'x1':1}}]&playerData={'level':23,'tutorialState':744,'gems':34234,'stars':12315345,'nmDigs':2,'&eggs':{'egg1':{'timeLeftToHatch':234,'breedType':34},{'egg2':{'timeLeftToHatch':234,'breedType':34},{'egg3':{'timeLeftToHatch':45,'breedType':12},{'egg4':{'timeLeftToHatch':234,'breedType':34}}
```

## Serialized Game State Hash

```
tunnelTown.php?qId=3&c=sd&b=0UHcm0C0Ggf%2fzrAI1F%2bR0FbQnu
4125c5MkipicWWDPOf%2fThNqh086wRwo4g2tAf89w48nGrSsJYnq0bD
NgvAzSxw%2fqmNkfLvlStn6uZxhVAsWy3JNR055QHxNyaTdGZt5b%2b
zTX5WTrEW1Luzt0cal%2f%2bU%2bcbpzei4IFoi0T10YOqxqdDfLRsli5Q
Gmcgyr%2blw2l%2f%2b4ZY51bdRkZiky7us%2bEmPxiJmMDajJKjAbb
433IIDG7GsUUvH4LPcBD39w8QVlhSBHylHBtSAUjtgpPFYSPAxyygAWJs
055%2bHr4K%2bklp1uVuMlffg5lx%2bD6zUc1qAh86q4hVhLM%2fqoo
YcXcZzCZCc1suH2pOckHuDc0RvKHNHHw2%2b8IFwNdSmRv9rzkeuNY
YIYjpUSknpGYfVWSK3R%2bACMvtvw8Cf%2fsoBjaQEmX3aW9LctQNEX
6Fb%2fJjgqu%2b5yimGa6VtV3HDRMiokwnPvHZqzQkJLASpnCoStodZ
BG9IGbSoaCWzvLmETW14m5pr69pZYVPO1R3sPCAwKQQ%2fpaEFxAH
q%2bliZPJYhHNcr1OfktoV35b3MLuh9fnYcxhMejNpevN2w5ly%2b%2fh
WhvTWXePr0ZiRJzam2DXCUwZVQ3B5ZDZLv9C1kwrPV1fRUoQvUH0%
2bRNXLW%2b0WfVabmhfZFufTKWpUgtyBXZiK2e6gSHXtH4G1tBK6d
F5lpxhHI4YZ17s8pZq%2bPrZxiWt1bUaXYjUFNjiLTyaam1TaopVONxsD
m0QkJbK297iKga09dMM76ff1hJgvyj%2ba55UcPxpBJUBjpQrAHIJHIJ
MnWc%2fvGiDLPJ7MjxJKpexF%2b2RbvzQfIYWF53wlfP9y8%2fZTQzGu
F3hlxScY%2b6JcynAKXdELK%2bVStkgi1KliNVSw08oyOgCp4LbfTa8HI
mlywU0o3aDum%2fGo60P%2bVZkHgoR5dS60VMe32LS9bbAsH%2f
ZaBgx1WA50tbH3bbiWvrz8cslfgjCVOsCeIO%2fJn4OQiBXNMCMOur2Ze
JCyo4XtJv6oeDIOJKCPAiPu4msaLaExtEMc3H5r8XSgS8bMoJFdkmKB
PCFSg2mvXglSQG6qXd6qBd%2bPVyQ5bDKKAPakS1apq69yA%2bBA
%2b2R2tsUNMhBkorCLNkD1RqgDiLt7KsHywnKtdSSAZFupQJN1%2f4fQ
```

## Obfuscation Differences



Y U SO HARD TO READ

# 1) Hashed Game State



- **Serialization - Google Protocol Buffers**

## Pros

- Less Server Code
- Fewer Database Schemas to Manage
- Obfuscation

## Cons

- Can be difficult for debugging
- Will need an External source for Metrics/Analytics

- **Game State versioning**

# Schema Migrations On The Client

```
private static GlobalState MigrateSaveState (GlobalState saveState)
{
    if(GetStateVersion(saveState) == OLD_VERSION_0_1)
    {
        saveState = UpdateStateFrom_0_1_To_0_2(saveState);
    }
    if(GetStateVersion(saveState) == OLD_VERSION_0_2)
    {
        saveState = UpdateStateFrom_0_2_To_0_3(saveState);
    }
    return saveState;
}

private static GlobalState UpdateStateFrom_0_1_To_0_2 (GlobalState saveState)
{
    foreach(InventoryItemObj invItem in inventoryList.serializableItems)
    {
        if(def.itemType == InventoryType.Plant)
        {
            invItem.ownedNumber = (invItem.ownedNumber <= 0) ? 1 : invItem.ownedNumber
        }
    }
    saveState.dataVersion = "0.2";
    return saveState;
}

private static GlobalState UpdateStateFrom_0_2_To_0_3 (GlobalState saveState)
{
    foreach(QuestState state in questStates.Values)
    {
        if(state.isCompleted) state.isCollected = true;
    }

    saveState.dataVersion = "0.3";
    return saveState;
}
```



# 1) Hashed Game State



- **Serialization - Google Protocol Buffers**

## Pros

- Less Server Code
- Fewer Database Schemas to Manage
- Obfuscation

## Cons

- Can be difficult for debugging
- Will need an External source for Metrics/Analytics

- **Game State versioning**

- **Storage friendly character sets**

## 2) Security

### Encryption of Game State

- Prevent share of save files
- Client needs to be able to decrypt without talking to server
- Don't leave keys/salt in plain text

### Server Authentication

- Confirming identity is important!



**TAKEAWAY:** If someone smart wants to hack your game, they will!

### 3) Server w/ Volatile Caching and Persistent Storage

Application language that can  
easily load balance



App server needs to be lean! Simple  
logic to allow easy scaling



Pick a database that will shard out  
of the box



Cache layer to let databases breathe!



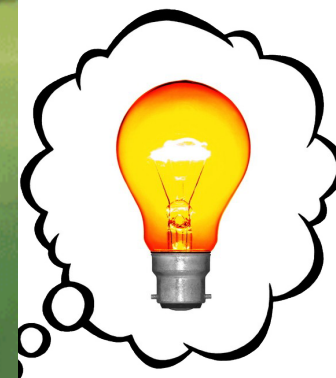


## 4) Online and Offline Play

How do you allow play on multiple devices while supporting online and offline modes?



# Houston, We Have a Problem



## 4) Online / Offline Formula

- Client always saves to disk and tries server
- Server responds with game state + UDID of last save it has

### Client Rules

- Server UDID == my UDID: use local
- Server UDID != my UDID:
  - If last save on this device was ONLINE: use server state
  - If last save on this device was OFFLINE: ask user



Device A

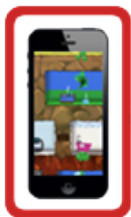


Persistent

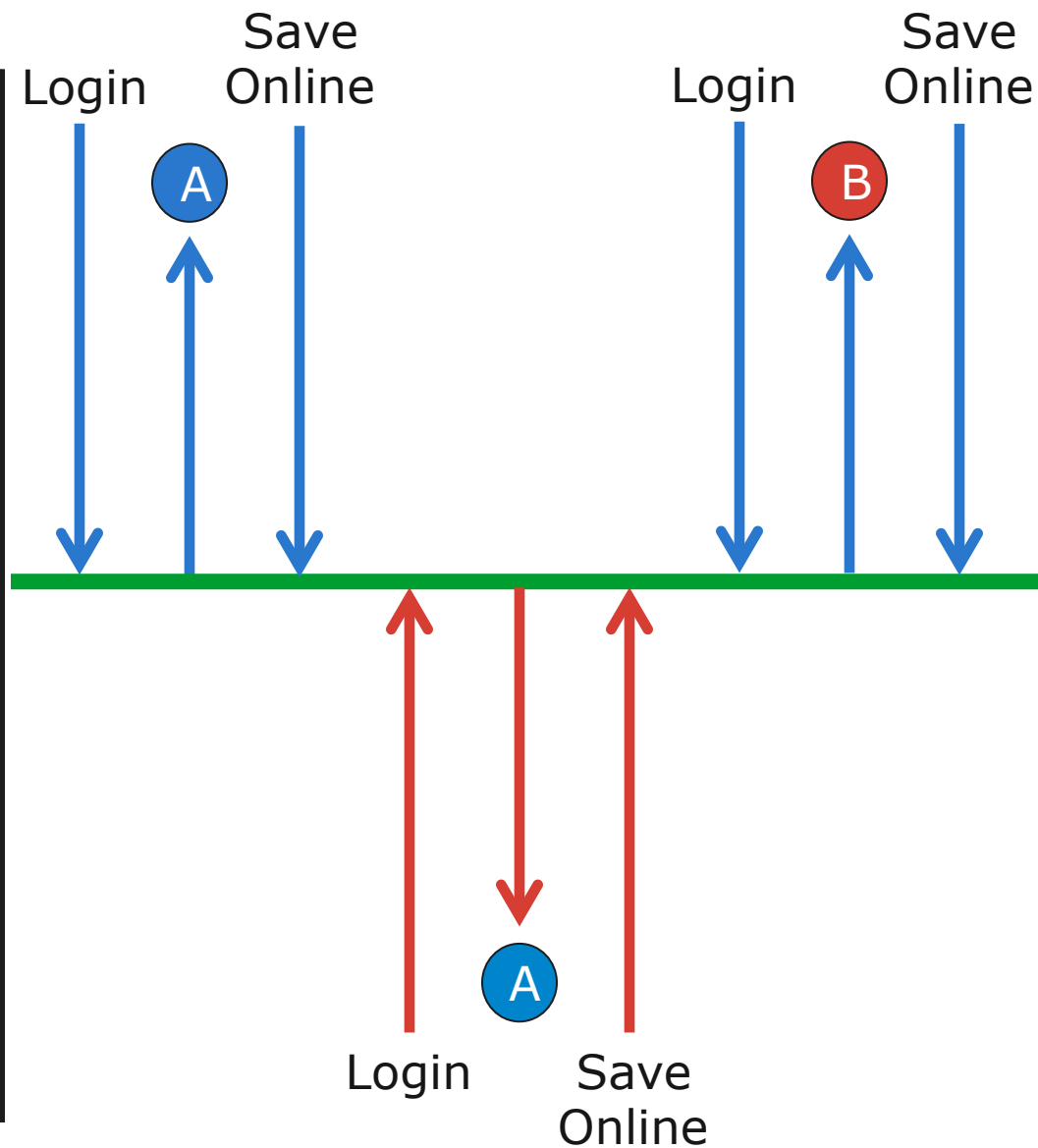


Storage

State: A



Device B





Device A



Persistent

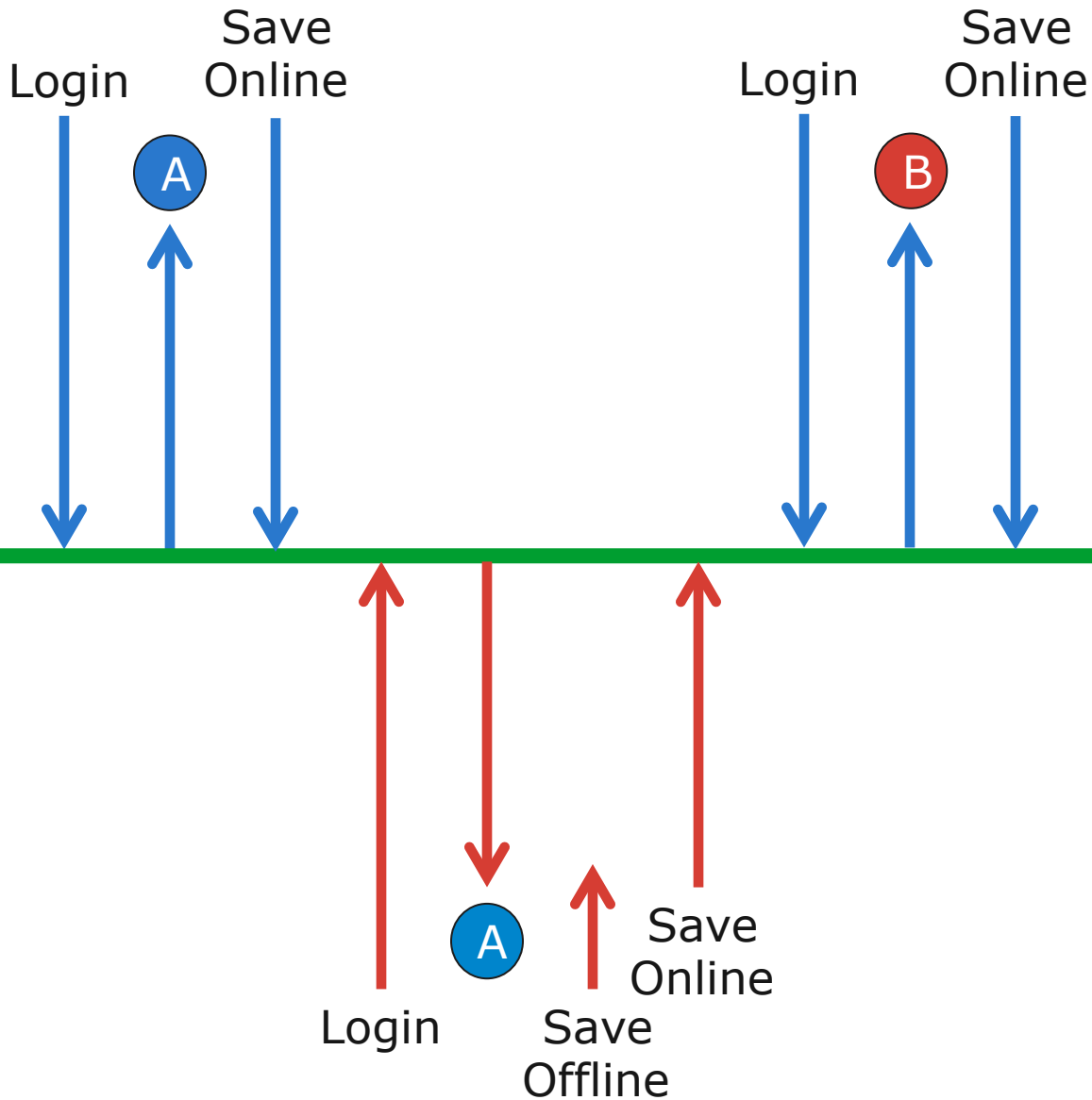


Storage

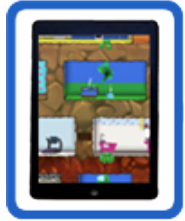
State: A



Device B



Device A



Persistent

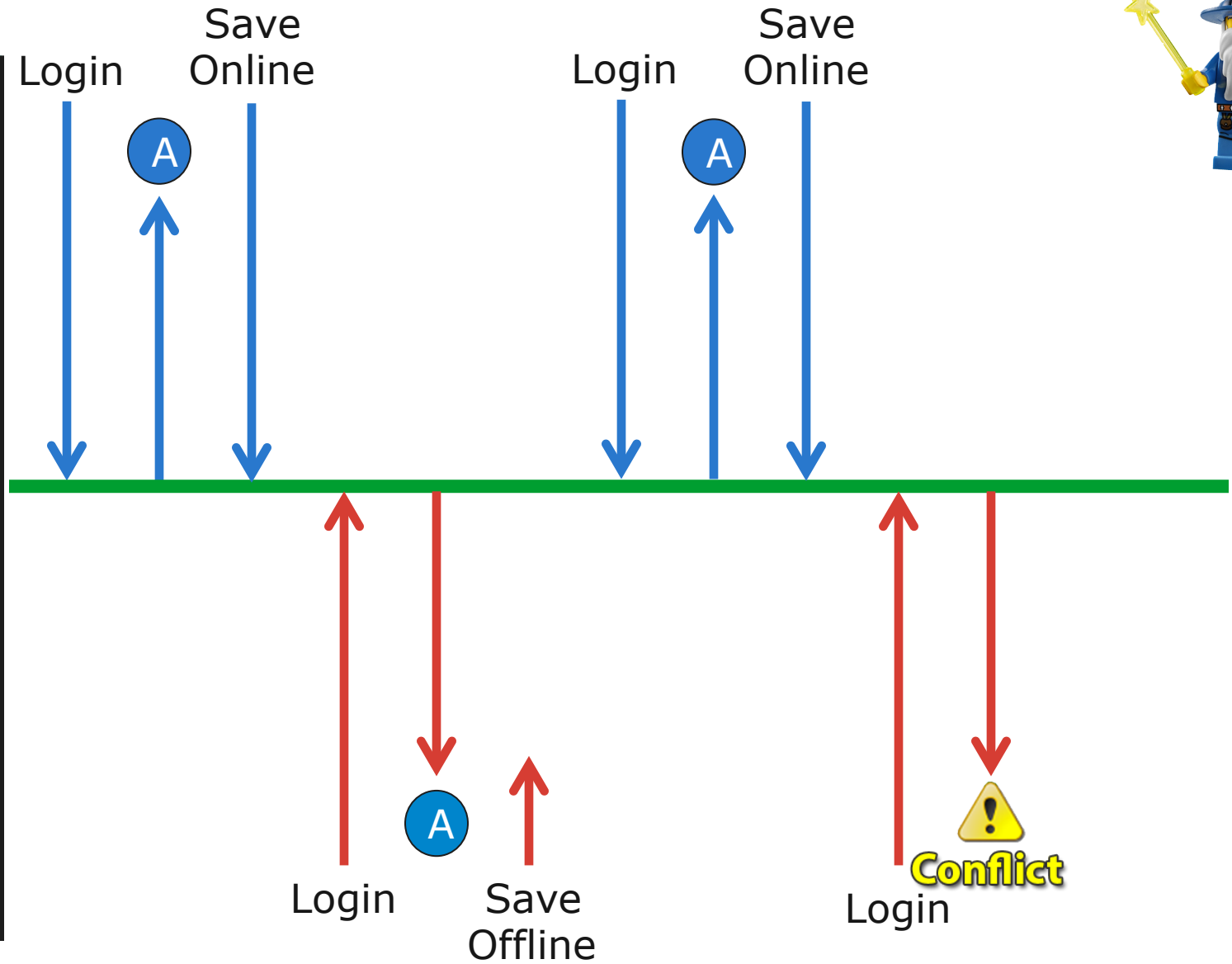


Storage

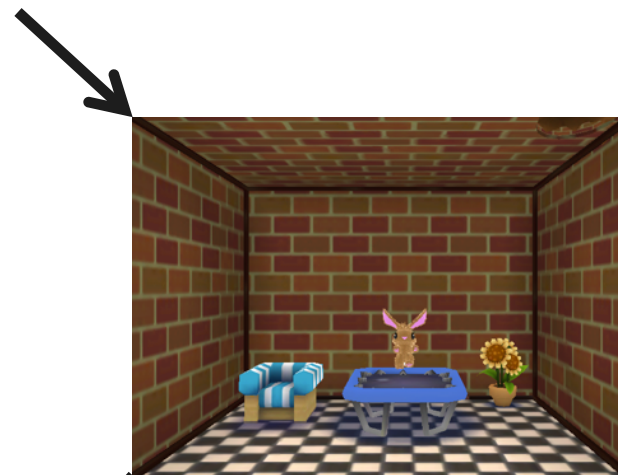
State: A



Device B



# Improvement: Merging



## 4) Same Time Play

**Problem:** playing on same account on multiple devices at the same time

### Potential Solutions

Don't Allow



Stomping



Stomp + Notify





## 4) Same Time Play

### Solution 1: Don't Allow

- Client sends heartbeat to server
- Client sends "I'm done" on pause/quit
- Server doesn't allow second device to play

#### Pro

1. No stomping

#### Cons

1. Requires server and client logic
2. Must ping servers frequently



## 4) Same Time Play

### Solution 2: Stomping

- Client gets game state from server on login
- Whoever saves last (closes app) wins

#### Pro

1. No added server or client code

#### Con

1. Stomping SUCKS

Users won't always realize they "stomped" their data and might think it's a bug if they "lost" data



## 4) Same Time Play

### Solution 3: Stomping w/ Notifications

- Client gets game state from server on login
- Whoever saves last (closes app) wins
- Notify users when stomping is detected

#### Pros

1. Users can keep playing
2. Notified if a stomp occurred



#### Cons

1. Stomping still SUCKS
2. Requires extra server/client code

# Hindsight Is Wonderful

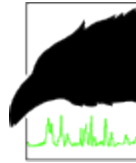
- People barely ever QUIT your app. Almost all “closes” are really just pauses
- Don't assume your target audience's hardware!
- Do research about silly reasons Apple has rejected apps!





## Server Tools

**Nagios®**



**MUNIN**



## Scaling Tools



CouchDB  
relax



Redis



Cassandra



Neo4j  
the graph database

mongoDB



membase



riak



## Client Tools



**Lumes**



**EDITOR CONSOLE PRO**



**U3DXT**

Questions?

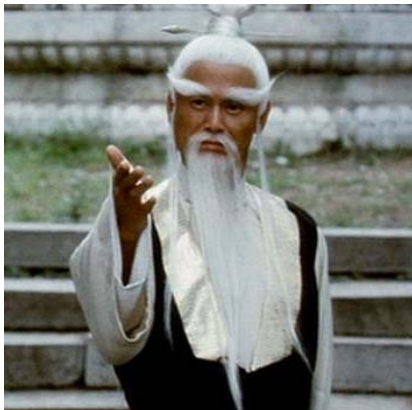


## Contact Information

Dave Geurts

daveg@smartbombinteractive.com

@sims11tz



Colter Haycock

colter@smartbombinteractive.com

@blindgoatia